IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received May 25, 2020, accepted June 4, 2020, date of publication June 8, 2020, date of current version June 25, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3000860

Data Modeling With Polynomial Representations and Autoregressive Time-Series Representations, and Their Connections

ASOKE K. NANDI, (Fellow, IEEE)
Department of Electronic and Computer Engineering, Brunel University London, Uxbridge UB8 3PH, U.K.
e-mail: asoke.nandi@brunel.ac.uk

ABSTRACT Two of the data modelling techniques - polynomial representation and time-series representation – are explored in this paper to establish their connections and differences. All theoretical studies are based on uniformly sampled data in the absence of noise. This paper proves that all data from an underlying polynomial model of finite degree q can be represented perfectly by an autoregressive time-series model of order q and a constant term μ as in equation (2). Furthermore, all polynomials of degree q are shown to give rise to the same set of time-series coefficients of specific forms with the only possible difference being in the constant term μ. It is also demonstrated that time-series with either non-integer coefficients or integer coefficients not of the aforementioned specific forms represent polynomials of infinite degree. Six numerical explorations, with both generated data and real data, including the UK data and US data on the current Covid-19 incidence, are presented to support the theoretical findings. It is shown that all polynomials of degree q can be represented by an all-pole filter with q repeated roots (or poles) at $z = +1$. Theoretically, all noise-free data representable by a finite order all-pole filter, whether they come from finite degree or infinite degree polynomials, can be described exactly by a finite order AR time-series; if the values of polynomial coefficients are not of special interest in any data modelling, one may use time-series representations for data modelling.

INDEX TERMS Data models, polynomials, autoregressive processes, time-series, signal representation, Covid-19.

I. INTRODUCTION

Interests in data science have been growing extremely fast in the twenty-first century. As well as interests from many different subject areas, data science is being integrated in diverse range of industries and agencies (e.g., health, transport, energy, government, society, etc.). Strictly, a time-series refers to a series of data points ordered in time. It is very common that a time-series represents data points at equally separated in time. Of course, the analytics that are created for time-series data can generally be applied to a sequence of data that are equally separated in space (e.g., images) or some other domain. There are many types of time-series models, including autoregressive models.

Although there are many types of time-series models, the earliest and an alternative way to model data is by polynomial regression. Polynomial regression models are generally fitted with the Least-squares method to obtain estimated values of the polynomial coefficients. In 1805 Legendre published the Least-squares method [1] and Gauss published it in 1809 and later in 1823 [2]. In 1815 Gergonne wrote a paper on "The application of the method of least squares to the interpolation of sequences" [3]. This is an English translation by Stigler [4] of the original paper that was written in French. In the last 120 or so years, polynomial regression contributed greatly to the development of regression analysis [5]–[7].

Although there are other ways to model data, the focus in this paper is around polynomial representation and autoregressive time-series representation. There has been a lot of research in time-series data representation [8]–[12]. For example, the main goal of time-series analysis in econometrics, geophysics, meteorology, quantitative finance, seismology, and statistics is prediction or forecasting [13]–[20]. On the other hand, it is used for signal detection and

The associate editor coordinating the review of this manuscript and approving it for publication was Datong Liu.

estimation in communication engineering, control engineering, and signal processing [21]–[28]. It is also used for clustering, classification, and prediction or forecasting in data mining, machine learning, and pattern recognition [29]–[34]. Mathematical modelling and time-series analysis are fundamental to many fields; a couple of very recent examples can be found in [35], [36].

In polynomial representations, observed data is a function of time (or some other variable). This function, except for the case of a constant or a straight line, represents a non-linear relationship between the time (or some other variable) and the observed data, even though the parameters are linear. On the other hand, in autoregressive (AR) time-series representation, observed data is a linear function of some of the earlier data and thus the model is linear in both data and parameters. Although both are used for data modelling, there are some fundamental differences. Hence, this paper explores many questions around polynomial and autoregressive representations with a view to establish their connections and differences. Two of these questions are:

1) Can all finite degree polynomials be expressed as finite order time series? If the answer is affirmative, what is the underlying relationship?

2) Can all finite order autoregressive time-series be represented as finite order polynomials?

This study is in the context of real-valued and uniformly sampled noise-free data. The paper presents the following original results:

1) All polynomials of degree 1 (linear), of degree 2 (quadratic), and of degree 3 (cubic) can be represented as autoregressive time-series of order 1, order 2, and order 3, with a constant respectively. This is illustrated in section II.
2) All polynomials of degree 3 can be represented by AR time-series with the set of coefficients with the same values but possibly with a different value for its constant term. This observation is also true for polynomials of degree 1 and of degree 2. This is presented in section II.
3) All polynomials of finite degree q can be represented as AR time-series of order q and a constant. This can be found in section III.
4) All polynomials of degree q can be represented by AR time-series with one set of coefficients with the same values but possibly with a different value for its constant term. This is demonstrated in section III.
5) The corresponding time-series coefficients are integers and of specific forms, which are derived in section III.
6) Some numerical explorations from several sources of both generated data and real data, including some current Covid-19 incidence data from the UK and the US, are presented in section IV.
7) Whilst all finite degree polynomials can be represented by finite order AR time-series, the converse is not true. There are infinitely many AR time-series of finite orders that cannot be represented by finite order polynomials. Furthermore, all finite order AR time-series with either non-integer coefficients or integer coefficients not of the aforementioned specific forms represent polynomials of infinite degree. This is shown in section V.
8) Section VI shows that all polynomials of degree q can be represented by an all-pole filter with q repeated roots (or poles) at $z = +1$. Thus, any noise-free data representable by a finite order all-pole filter, whether they come from finite degree or infinite degree polynomials, can be described exactly by a finite order AR time-series.

II. METHOD – SMALL DEGREE POLYNOMIAL

Given a set of uniformly sampled real-valued data points in discrete time, these may be represented by a polynomial or a time-series. A polynomial of degree N in continuous time can take the following form

$$y(t) = \sum_{i=0}^{N} c(i)\, t^{i}$$

For uniformly sampled discrete time, the continuous time, t, is represented as $t = nT$, where n is an integer and T is the sampling period. In this scenario, the above equation can be rewritten as

$$y(nT) = \sum_{i=0}^{N} c(i)\,(nT)^{i} \tag{1}$$

On the other hand, an autoregressive time-series model of order q, AR(q), can be written as

$$y(n) = \sum_{i=1}^{q} a(i)\, y(n-i) + \mu \tag{2}$$

and may be used to represent the set of uniformly sampled data points in discrete time.

A. LINEAR POLYNOMIAL

In this subsection, an exploration of data representation by a linear polynomial and an AR time-series is carried out. For any linear polynomial, N has the value of 1 in equation (1). It is easy to show from equation (1) that $y(nT) = y(nT - T) + c(1)\,T$. By removing T from indices, this can be written as $y(n) = y(n-1) + c(1)\,T$. Comparing this with equation (2) for AR(q), it is clear that $q = 1$, $a(1) = 1$, and $\mu = c(1)\,T$.

Therefore, the following can be concluded:

- Every linear polynomial, i.e., of degree 1, can be perfectly represented by an AR(1) time-series.
- Every linear polynomial will have the same value of the coefficient in time-series, i.e., $a(1) = 1$.
- The constant term in the time-series is given by $\mu = c(1)\,T$.
- This implies that every linear polynomial with different values of $c(0)$ but the same value of $c(1)$ will have the identical AR(1) representation, i.e., with the same values of $a(1)$ and μ.

B. QUADRATIC POLYNOMIAL

In this subsection, an exploration of data representation by a quadratic polynomial and an AR time-series is carried out. For any quadratic polynomial, N has the value of 2 in equation (1). Thus, it follows from equation (1) that

$$y(nT) = c(0) + c(1)\,(nT) + c(2)\,(nT)^2, \tag{3}$$

$$y(nT-T) = c(0) + c(1)\,(nT-T) + c(2)\,(nT-T)^2, \tag{4}$$

$$y(nT-2T) = c(0) + c(1)\,(nT-2T) + c(2)\,(nT-2T)^2, \tag{5}$$

Using equations (3) and (4), one can write

$$y(nT) = y(nT-T) + c(1)\,T + 2c(2)\,nT^2 - c(2)\,T^2, \tag{6}$$

and, using equations (4) and (5), one can write

$$y(nT-T) = y(nT-2T) + c(1)\,T + 2c(2)\,nT^2 - 3c(2)\,T^2, \tag{7}$$

Now, using equations (6) and (7), one finds

$$y(nT) = y(nT-T) + [y(nT-T) - y(nT-2T)] + 2c(2)\,T^2, \tag{8}$$

Therefore,

$$y(nT) = 2y(nT-T) - y(nT-2T) + 2c(2)\,T^2 \tag{9}$$

By removing T from indices, equation (9) can be written as $y(n) = 2y(n-1) - y(n-2) + 2c(2)\,T^2$. Comparing this with equation (2) for AR(q), it is clear that $q = 2$, $a(1) = 2, a(2) = -1$, and $\mu = 2c(2)\,T^2$.

Therefore, the following can be concluded:

- Every quadratic polynomial, i.e., of degree 2, can be perfectly represented by an AR(2) time-series.
- Every quadratic polynomial will have the same coefficient values in time-series, i.e., $a(1) = 2$ and $a(2) = -1$.
- The constant term in the time-series is given by $\mu = 2c(2)\,T^2$.
- This implies that every quadratic polynomial with different values of $c(0)$ and $c(1)$ but the same value of $c(2)$ will have the identical AR(2) representation, i.e., with the same values of $a(1)$, $a(2)$, and μ.

C. CUBIC POLYNOMIAL

In this subsection, an exploration of data representation by a cubic polynomial and an AR time-series is carried out. For any cubic polynomial, N has the value of 3 in equation (1). Thus, it follows from equation (1) that

$$y(nT) = c(0) + c(1)\,(nT) + c(2)\,(nT)^2 + c(3)\,(nT)^3, \tag{10}$$

$$y(nT-T) = c(0) + c(1)\,(nT-T) + c(2)\,(nT-T)^2 + c(3)\,(nT-T)^3, \tag{11}$$

$$y(nT-2T) = c(0) + c(1)\,(nT-2T) + c(2)\,(nT-2T)^2 + c(3)\,(nT-2T)^3, \tag{12}$$

$$y(nT-3T) = c(0) + c(1)\,(nT-3T) + c(2)\,(nT-3T)^2 + c(3)\,(nT-3T)^3, \tag{13}$$

Using equations (10) and (11), one can obtain

$$y(nT) = y(nT-T) + c(1)\,T + c(2)\,2nT^2 - c(2)\,T^2 + c(3)\,3n^2T^3 - c(3)\,3nT^3 + c(3)\,T^3, \tag{14}$$

and, using equations (11) and (12), one can obtain

$$y(nT-T) = y(nT-2T) + c(1)\,T + c(2)\,2nT^2 - c(2)\,3T^2 + c(3)\,3n^2T^3 - c(3)\,9nT^3 + c(3)\,7T^3, \tag{15}$$

Now, using equations (14) and (15), one obtains

$$y(nT) - y(nT-T) = y(nT-T) - y(nT-2T) + 2c(2)\,T^2 + 6c(3)\,nT^3 - 6c(3)\,T^3, \tag{16}$$

Using equations (12) and (13), one can write

$$y(nT-2T) = y(nT-3T) + c(1)\,T + c(2)\,2nT^2 - c(2)\,5T^2 + c(3)\,3n^2T^3 - c(3)\,15nT^3 + 19c(3)\,T^3, \tag{17}$$

Now, using equations (15) and (17), one can write

$$y(nT-T) - y(nT-2T) = y(nT-2T) - y(nT-3T) + 2c(2)\,T^2 + 6c(3)\,nT^3 - 12c(3)\,T^3, \tag{18}$$

Thus,

$$y(nT-T) - 2y(nT-2T) + y(nT-3T) + 6c(3)\,T^3 = 2c(2)\,T^2 + 6c(3)\,nT^3 - 6c(3)\,T^3, \tag{19}$$

Combining equations (16) and (19), one obtains

$$y(nT) - y(nT-T) = y(nT-T) - y(nT-2T) + y(nT-T) - 2y(nT-2T) + y(nT-3T) + 6c(3)\,T^3,$$

Therefore,

$$y(nT) = 3y(nT-T) - 3y(nT-2T) + y(nT-3T) + 6c(3)\,T^3 \tag{20}$$

By removing T from indices, this can be written as $y(n) = 3y(n-1) - 3y(n-2) + y(n-1) + 6c(3)\,T^3$. This can be described by AR(q), provided $q = 3$, $a(1) = 3, a(2) = -3, a(3) = 1$, and $\mu = 6c(3)\,T^3$.

TABLE 1. Information for polynomials of different degrees.

Degree of polynomial N	AR parameters			
	$a(1)$	$a(2)$	$a(3)$	μ
1	1			$c(1)T$
2	2	-1		$2c(2)T^2$
3	3	-3	1	$6c(3)T^3$

Therefore, the following can be concluded:

- Every cubic polynomial, i.e., of degree 3, can be perfectly represented by an AR(3) time-series.
- All cubic polynomials will have the same coefficient values in time-series, i.e., $a(1) = 3$, $a(2) = -3$, and $a(3) = 1$.
- The constant term in the time-series is given by $\mu = 6c(3)\,T^3$.
- This implies that every cubic polynomial with different values of $c(0)$, $c(1)$, and $c(2)$ but the same value of $c(3)$ will have the identical AR(3) representation, i.e., with the same values of $a(1)$, $a(2)$, $a(3)$, and μ.

The summary of the exposition so far is that all polynomials of degree 1 (linear), of degree 2 (quadratic), and of degree 3 (cubic) can be perfectly represented as AR time-series of orders 1, 2, and 3 respectively. Furthermore, for each degree of polynomials all the time-series coefficients have predefined values and they are specific integers, while the constant term, μ, has a predefined form that depends on the coefficient of the leading degree of the polynomial, the degree of the polynomial, and the sampling period. These and more specific information can be found in Table 1 above.

III. METHOD – ANY FINITE DEGREE POLYNOMIAL

In section II it has been demonstrated that all polynomials of degree 1 (linear), of degree 2 (quadratic), and of degree 3 (cubic) can be perfectly represented as autoregressive time-series of orders 1, 2, and 3 respectively. In this section the exploration is generalised for all polynomials of every finite degree. In section II it was found that, for $q = 1, 2,$ and 3, the degree of the polynomial and the corresponding order of the AR time-series order are identical. In the following, a discrete-time polynomial of degree q of the form below is considered

$$y\,(nT) = \sum_{j=0}^{q} c\,(j)\,(nT)^j \tag{21}$$

in seeking a corresponding autoregressive time-series model of order q, AR(q).

The time-series in equation (2) can be rewritten as

$$y\,(n) - \mu = \sum_{i=1}^{q} a\,(i)\,y\,(n-i) \tag{22}$$

Now it is conjectured that

$$a\,(i) = (-1)^{i+1}\binom{q}{i} \tag{23}$$

for $i = 1, 2, \ldots, q$. Using this conjecture and equation (21), the equation (22) can be written as

$$y\,(nT) - \mu = \sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}\sum_{j=0}^{q} c\,(j)\,(nT - iT)^j \tag{24}$$

In the above double summation, it is instructive and revealing to consider different values of j separately.

A. PART I

For the particular case of $j = 0$, the right-hand side of equation (24) can be written as $\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}c(0)$. The relation 0.154.6 on page 4 of [37], for $q \geq n \geq 1$ and $0^0 \equiv 1$, can be adapted to

$$\sum_{i=0}^{q}(-1)^i\binom{q}{i}i^{n-1} = 0 \tag{25}$$

Using this relation, for $q \geq n = 1$,

$$\begin{aligned}\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i} &= \sum_{i=0}^{q}(-1)^{i+1}\binom{q}{i} + 1\\ &= -\sum_{i=0}^{q}(-1)^{i}\binom{q}{i} + 1 = 1\end{aligned} \tag{26}$$

Therefore, for the case of $j = 0$, the right-hand side of equation (24) is $c\,(0)$.

B. PART II

For the case of $j = 1$, the right-hand side of equation (24) can be written as

$$\begin{aligned}&\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}\{c\,(1)\,(nT - iT)\}\\ &= c\,(1)\,nT\sum_{i=1}^{q}(-1)^{i+i}\binom{q}{i}\\ &- c\,(1)\,T\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i\end{aligned} \tag{27}$$

Using equation (25), for $q \geq n > 1$, one can write

$$\begin{aligned}\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i^{n-1} &= \sum_{i=0}^{q}(-1)^{i+1}\binom{q}{i}i^{n-1}\\ &= -\sum_{i=0}^{q}(-1)^{i}\binom{q}{i}i^{n-1} = 0\end{aligned} \tag{28}$$

Using equations (26) and (28) in equation (27), it is found that the right-hand side of equation (24), for the case of $j = 1$, is equal to $c\,(1)\,nT$.

C. PART III

Now the case of $j = 2$ is considered. The right-hand side of equation (24) can be written as

$$\begin{aligned}&\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}\left\{c\,(2)\,(nT - iT)^2\right\}\\ &= c\,(2)\,(nT)^2\sum_{i=1}^{q}(-1)^{i+i}\binom{q}{i}\end{aligned}$$

$$
\begin{aligned}
&-2c\,(2)\,nT^2\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i\\
&+c\,(2)\,(-T)^2\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i^2
\end{aligned}
$$

Using equations (26) and (28), in the previous expression for the right-hand side of equation (24), for the case of $j = 2$, the right-hand side is found to be equal to $c\,(2)\ \ (nT)^2$.

Similarly, for each value of j up to $j = q - 1$, it can be shown that the right-hand side of equation is equal to $c\,(j)\,(nT)^j$. When $j = q - 1$, there is a term of the form $c\,(q-1)\,(-T)^{q-1}\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i^{q-1}$. According to equation (28), which is valid if the top range of the summation is either larger than or equal to the power of i plus one, i.e., $q \geq (q-1)+1$, this term equates to zero.

However, when $j = q$, there is a term of the form $c\,(q)\,(-T)^q\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i^q$. For this term, equation (28) is not valid since the top range of the summation, i.e., q, is neither larger than nor equal to the power of i plus one, i.e., $(q+1)$. To deal with the case of $j = q$, the relation 0.154.4 on page 4 of [37], for $n \geq 0$ and $0^0 \equiv 1$, can be adapted to

$$\sum_{i=0}^{q}(-1)^i\binom{q}{i}i^q = (-1)^q\,q! \tag{29}$$

Thus,

$$
\begin{aligned}
\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i^q &= \sum_{i=0}^{q}(-1)^{i+1}\binom{q}{i}i^q\\
&= -\sum_{i=0}^{q}(-1)^{i}\binom{q}{i}i^q = -\,(-1)^q\,q!
\end{aligned}
\tag{30}
$$

Thus, for the case of $j = q$, the right-hand side of equation (24) can be written as

$$
\begin{aligned}
&\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}\left\{c\,(q)\,(nT-iT)^q\right\}\\
&= c\,(q)\,(nT)^q\sum_{i=1}^{q}(-1)^{i+i}\binom{q}{i}+\ldots+c\,(q)\,(-T)^q\\
&\quad\times\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}i^q
\end{aligned}
$$

Using equation (26) the first term is $c\,(q)\,(nT)^q$. All the terms in the middle are zero by virtue of equation (28). Using equation (30) the last term is found to be $c\,(q)\,(-T)^q\,(-\,(-1)^q\,q!)$, which is equal to $-c\,(q)\,T^q(q!)$.

Adding all the results for $j = 0, 1, \ldots, q$, one obtains

$$
\begin{aligned}
y\,(nT)-\mu &= \sum_{j=0}^{q}c\,(j)\ \ (nT)^j - c\,(q)\,T^q\,(q!)\\
&= y\,(nT) - c\,(q)\,T^q\,(q!)
\end{aligned}
$$

Thus,

$$\mu = c\,(q)\,T^q\,(q!) \tag{31}$$

Therefore, all noise-free data from uniformly sampled polynomials of finite degree q can be perfectly represented by an autoregressive time-series model of order q such that

$$y\,(n) = \sum_{i=1}^{q}a\,(i)\,y\,(n-i)+\mu$$

where

$$\mu = c\,(q)\,T^q\,(q!)$$

and

$$a\,(i) = (-1)^{i+1}\binom{q}{i},\quad \text{for } i = 1, 2, \ldots, q$$

IV. EXPERIMENTS

In this section some explorations are carried out for different types of data sources to illustrate a few themes. In reality, all real data have uncertainties; therefore, it is important to study sensitivities to degrees and types of uncertainties. Yet, in these explorations all generated data are error-free. Here the objectives are to underpin some theoretical results and to generate some intuitions from precise data and theoretical results, and not to get distracted into studying effects of noise interference. Two applications to real data, the current Covid-19 data from the UK and the US, are clearly not noise-free but are offered as real examples.

In the first four of these explorations, N data are generated. These are then modelled by polynomials as in equation (21) and time-series as in equation (2). When considering a polynomial of degree p, the first $(p+1)$ data are used to evaluate the $(p+1)$ coefficients of this polynomial. This works as data are error-free. On the other hand, when considering a time-series of order q, the first $2q$ data are used to evaluate the q coefficients of this time-series.

A. CASE I

Here N data are generated from a polynomial of degree 3,

$$
\begin{aligned}
y\,(n) = (nT)^3 - 2\,(nT)^2\\
+\,(nT)-1,\ \text{for } T = 0.2 \quad \text{and } n = -17:1:17.
\end{aligned}
$$

This is a finite degree polynomial with no steady state. For each value of the degree p of the polynomial from $p = 1, \ldots, 9$, the first $(p+1)$ data are used to calculate the $(p+1)$ coefficients of the polynomial. Using these polynomial coefficients, the remaining $(N-p-1)$ data values are predicted; these are labelled as $yp(n)$ for $n = (p+2), \ldots, N$. Similarly, for each value of the time-series order of q from $q = 1, \ldots, 9$, the first $2q$ data are used to calculate the q coefficients of the time-series. Using these coefficients, the remaining $(N-2q)$ data values are predicted; these are labelled as $yt(n)$ for $n = (2q+1), \ldots, N$.

For the same values of $(p+1)$ and q, $(N-p-1)$ data values are predicted for polynomial and $(N-2q)$ data

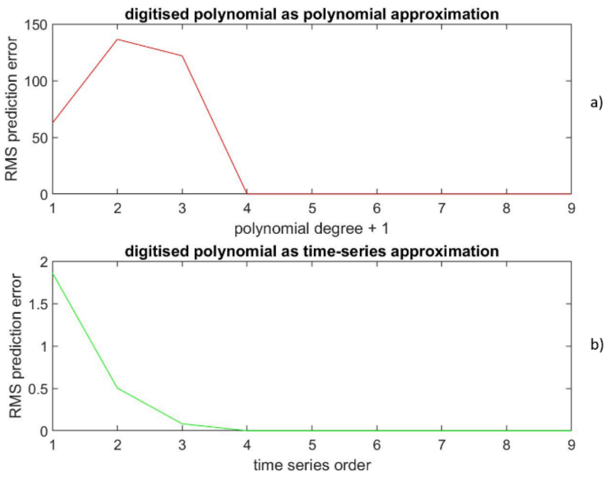

FIGURE 1. Data generated from a polynomial, Figure 1a) shows the RMS prediction error (polynomial) as a function of $(p + 1)$. Figure 1b) presents the RMS prediction error (time-series) as a function of q.

values are predicted for time-series representations. To compare prediction errors from polynomial and time-series representations fairly, only those predictions, i.e., $(N-2q)$ data values, common to both representations are used. Mean prediction errors are $\left(\sum_{i=2q+1}^{N}(yp\,(i)-y\,(i))\right)/(N-2q)$ and $\left(\sum_{i=2q+1}^{N}(yt\,(i)-y\,(i))\right)/(N-2q)$ for polynomial and time-series respectively. Also, the RMS prediction error (polynomial) is defined as

$$\sqrt{\left[\left(\sum_{i=2q+1}^{N}(yp\,(i)-y\,(i))^2\right)/(N-2q)\right]}$$

while the RMS prediction error (time-series) is defined as

$$\sqrt{\left[\left(\sum_{i=2q+1}^{N}(yt\,(i)-y\,(i))^2\right)/(N-2q)\right]}$$

The RMS prediction error (polynomial) is depicted in Figure 1a) as a function of $(p+1)$, while the RMS prediction error (time-series) is shown in Figure 1b) as a function of q. The prediction error at $(p+1) = 4$ is $(6.6 * 10^{-13} \pm 2.7 * 10^{-12})$, while the prediction error at $q = 4$ is $\left(7.2 * 10^{-10} \pm 1.6 * 10^{-9}\right)$; both are extremely small. Figure 2a) shows the data versus the time index, while the Figure 2b) depicts the prediction errors versus the time index for $(p+1) = 4$ (polynomial in red) and at $q = 4$ (time-series in green). The results confirm that these data from a finite degree polynomial can be equally well described by both polynomial and time-series representations.

B. CASE II

Here N data are generated from a sine wave

$$y\,(n) = sin(2\pi n/16), \quad \text{for } n = -17:1:17.$$

This represents an infinite degree polynomial and has no steady state, but its values are bounded between -1 and +1.

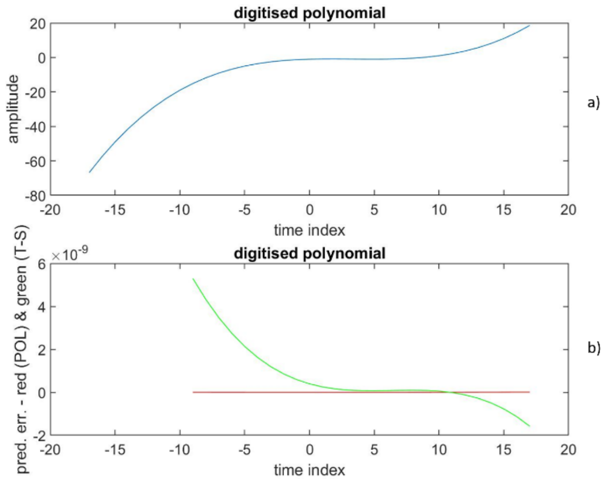

FIGURE 2. Data generated from a polynomial, Figure 2a) depicts the data versus the time index. Figure 2b) shows the prediction errors versus the time index for $(p + 1) = 4$ (polynomial in red) and for $q = 4$ (time-series in green).

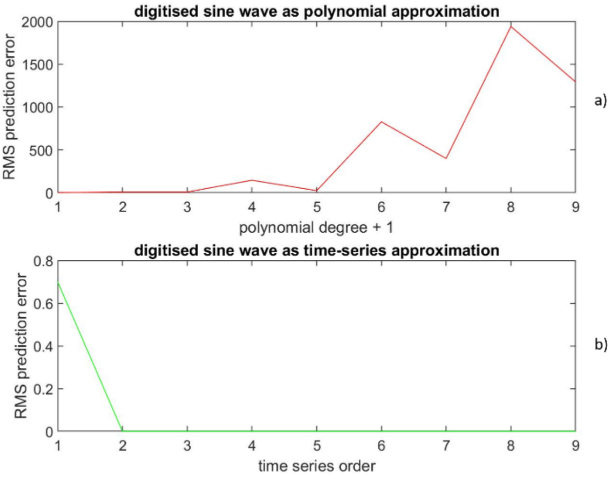

FIGURE 3. Data generated from a sine wave, Figure 3a) presents the RMS prediction error (polynomial) as a function of $(p + 1)$. Figure 3b) displays the RMS prediction error (time-series) as a function of q.

The procedures for calculating the $(p+1)$ coefficients of the polynomial and calculating the q coefficients of the AR time-series are the same as described in Case I earlier. Also, the procedures for calculating the prediction error (polynomial) and the prediction error (times-series) have been described earlier in Case I.

The RMS prediction error (polynomial) is depicted in Figure 3a) as a function of $(p+1)$, while the RMS prediction error (time-series) is shown in Figure 3b) as a function of q. The error at $(p+1) = 1$ is (6.9 ± 3.8), while the error at $q = 2$ is $\left(2.5 * 10^{-18} \pm 1.4 * 10^{-15}\right)$. Also, the error at $(p+1) = 6$ is (422 ± 614), while the error at $q = 2$ is $\left(1.3 * 10^{-16} \pm 1.4 * 10^{-15}\right)$. Figure 4a) shows the data versus the time index, while the Figure 4b) depicts the prediction errors versus the time index for $(p+1) = 1$ (polynomial in red) and at $q = 2$ (time-series in green). The results confirm

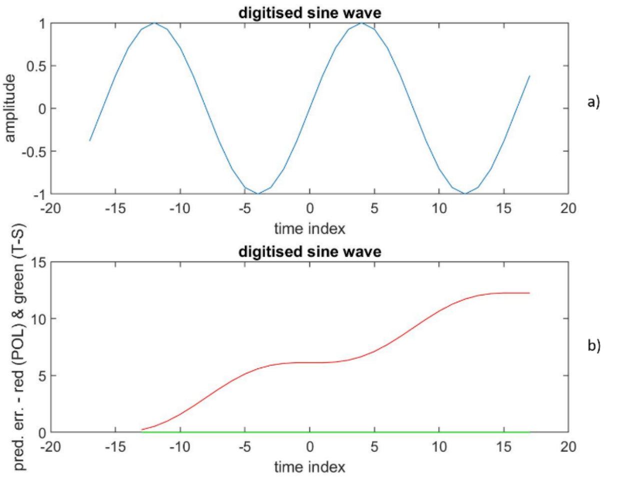

FIGURE 4. Data generated from a sine wave, Figure 4a) shows the data versus the time index. Figure 4b) depicts the prediction errors versus the time index for $(p+1) = 1$ (polynomial in red) and for $q = 2$ (time-series in green).

that these data from a sine wave are extremely well described by a time-series representation of only order 2; there is a theoretical reason for this (see section V for an explanation). Also, this time series representation is far better than any finite degree polynomial representation.

C. CASE III

Here N data are generated from a non-polynomial

$$y(n) = 1 - 2n - 3n(n-1) + (0.5)^n, \quad \text{for } n = -17 : 1 : 17.$$

This represents an infinite degree polynomial and has no steady state. The procedures for calculating the $(p+1)$ coefficients of the polynomial and calculating the q coefficients of the time-series are the same as described in Case I earlier. Also, the procedures for calculating the prediction error (polynomial) and the prediction error (times-series) have been described earlier in Case I.

The RMS prediction error (polynomial) is depicted in Figure 5a) as a function of $(p+1)$, while the RMS prediction error (time-series) is shown in Figure 5b) as a function of q. The prediction error at $(p+1) = 3$ is $(6.6 * 10^6 \pm 4.9 * 10^6)$, while the prediction error at $q = 4$ is $\left(-1.2 * 10^{-8} \pm 8.7 * 10^{-8}\right)$. Also, the prediction error at $(p+1) = 5$ is $(7.3 * 10^7 \pm 8.9 * 10^7)$, while the prediction error at $q = 4$ is $\left(-1.2 * 10^{-8} \pm 8.7 * 10^{-8}\right)$. Figure 6a) shows the data versus the time index, while the Figure 6b) depicts the prediction errors versus the time index for $(p+1) = 5$ (polynomial in red) and at $q = 4$ (time-series in green). Thus, these data from a non-polynomial are significantly better described by a time-series representation of only order 4; the theoretical reason can be found in section V. Also, RMS (time-series) is many orders of magnitude smaller than RMS (any finite degree polynomial).

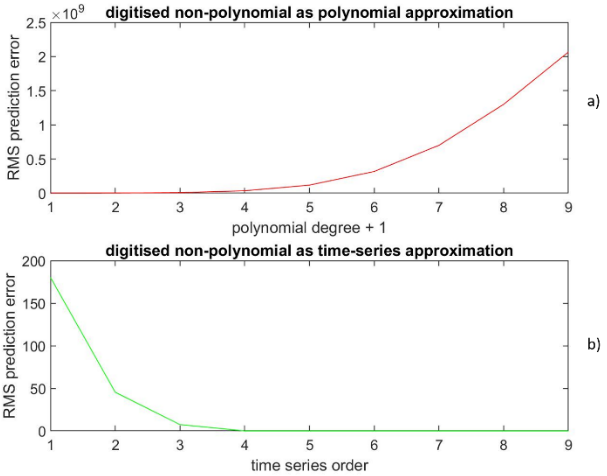

FIGURE 5. Data generated from a non-polynomial, Figure 5a) presents the RMS prediction error (polynomial) as a function of $(p+1)$. Figure 5b) displays the RMS prediction error (time-series) as a function of q.

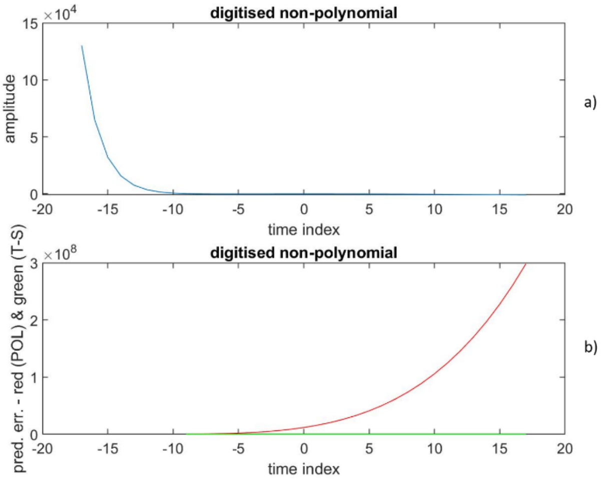

FIGURE 6. Data generated from a non-polynomial, Figure 6a) shows the data versus the time index. Figure 6b) depicts the prediction errors versus the time index for $(p+1) = 5$ (polynomial in red) and for $q = 4$ (time-series in green).

D. CASE IV

Here N data are generated from an inverse polynomial

$$y(n) = 1/(1 + 0.2n), \quad \text{for } n = 0 : 1 : 34.$$

This represents an infinite degree polynomial. It has neither a finite degree polynomial representation nor a finite order time-series representation. The procedures for calculating the $(p+1)$ coefficients of the polynomial and calculating the q coefficients of the time-series are the same as described in Case I earlier. Also, the procedures for calculating the prediction error (polynomial) and the prediction error (times-series) have been described earlier in Case I.

The RMS prediction error (polynomial) is depicted in Figure 7a) as a function of $(p+1)$, while the RMS prediction error (time-series) is shown in Figure 7b) as a function of q.

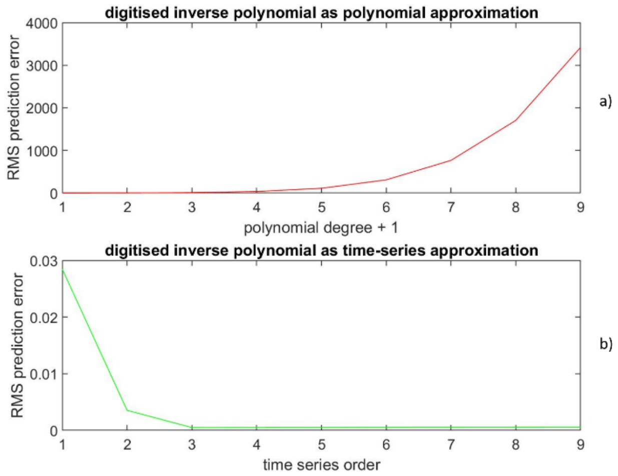

FIGURE 7. Data generated from an inverse polynomial, Figure 7a) displays the RMS prediction error (polynomial) as a function of $(p+1)$. Figure 7b) presents the RMS prediction error (time-series) as a function of q.

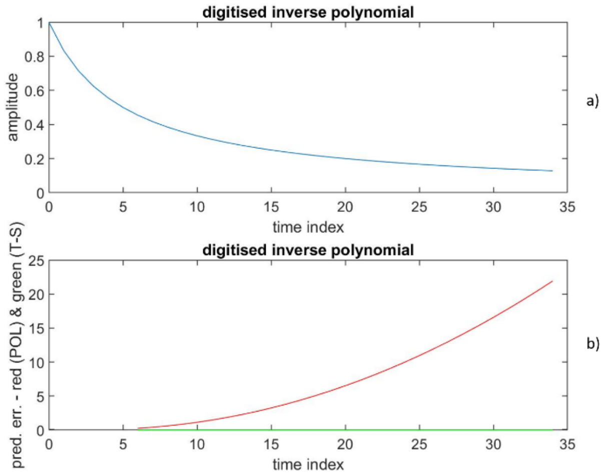

FIGURE 8. Data generated from an inverse polynomial, Figure 8a) shows the data versus the time index. Figure 8b) depicts the prediction errors versus the time index for $(p+1)=3$ (polynomial in red) and for $q=3$ (time-series in green).

The prediction error at $(p+1)=3$ is (8.2 ± 6.7), while the prediction error at $q=3$ is $(-4.7*10^{-4} \pm 1.7*10^{-4})$. Also, the prediction error at $(p+1)=7$ is (476 ± 617), while the prediction error at $q=7$ is $(-1.1*10^{-7} \pm 9.5*10^{-8})$. Figure 8a) shows the data versus the time index, while the Figure 8b) depicts the prediction errors versus the time index for $(p+1)=3$ (polynomial in red) and at $q=3$ (time-series in green). Results confirm that these data from an inverse polynomial are significantly better described by an AR time-series representation than a finite degree polynomial representation by several orders of magnitude in RMS.

E. CASE V

This is an example of using real data from a current global Covid-19 epidemic as it is unfolding. The dataset represents cumulative daily confirmed cases of Covid-19 infections in the UK. This dataset is publicly available [38]. On 01 April 2020 there were 61 data (i.e., N = 61) covering the period from 31 January 2020 to 31 March 2020. Thus $y(n)$ for $n = 1:1:61$ represents the cumulative daily confirmed cases of Covid-19 infections in the UK.

Of these 61 data, the first 50 data are used for estimating the free parameters and the last 11 data are used for forecasting. For a polynomial of the degree p, the first 50 data are used to estimate the $(p+1)$ coefficients of the polynomial using the Moore-Penrose inverse. By adopting the equation (21), the first 50 data can be described the matrix equation $Y = ACp$, where

$$\begin{aligned} Y &= [y(1)\, y(2) \ldots y(50)]^{T} \\ Cp &= [c(0)\, c(1) \ldots c(p)]^{T} \\ A &= [11\ldots 1;\, 12\ldots 2^{p};\, \ldots;\, 150\ldots 50^{p}] \end{aligned}$$

Thus, Y is a column vector of size 50×1, C is a column vector of size $(p+1)x1$, and A is a matrix of size $50x(p+1)$. Now,

$$Cp = \left(A^{T}A\right)^{-1} A^{T}Y \tag{32}$$

Using these estimated polynomial coefficients from the equation (32), all 61 data are calculated using

$$Yp = BCp \tag{33}$$

where B is a matrix of size $61x(p+1)$ and $B = [11\ldots 1;\, 12\ldots 2^{p};\, \ldots;\, 161\ldots 61^{p}]$, while Yp is a column vector of size 61×1 and $Yp = [yp(1)\, yp(2) \ldots yp(61)]^{T}$. Of course, $[yp(1)\, yp(2) \ldots yp(50)]^{T}$ came from the regression but $[yp(51)\, yp(52) \ldots yp(61)]^{T}$ are polynomial predictions for $[y(51)\, y(52) \ldots y(61)]^{T}$.

Similarly, for a time-series of order q, the first 50 data are used to estimate the q coefficients of the time-series. Each of these data values depends on the coefficients and earlier data values. As all data values are error prone, the Total Least Squares, which takes account of errors in both the dependent and independent variables, is more appropriate than the ordinary Least Squares, which takes account of only errors in dependent variables and not in the independent variables. Using the q estimated coefficients, all 61 data are calculated, which are labelled as $YT = [yt(1)\, yt(2) \ldots yt(61)]^{T}$. Of course, out of these 61 values, $[yt(1)\, yt(2) \ldots yt(50)]^{T}$ came from the regression and $[yt(51)\, yt(52) \ldots yt(61)]^{T}$ are time-series predictions for $[y(51)\, y(52) \ldots y(61)]^{T}$.

It is not known a priori whether the data can be represented by a finite degree polynomial or a finite order time-series. The RMS error at $(p+1)=5$ is 5142, while the RMS error at $q=2$ is 539. Clearly, the time-series representation is much more accurate. Also, the RMS error at $(p+1)=6$ is 1711, much smaller than at lower p values, but it is still much larger than the one from the time-series representation. Figure 9a) depicts all 61 data values (y) in blue, all 61 calculated values (yp) in red according to polynomial representation at $(p+1)=6$ [the first 50 values are from the fit and the last 11 values are predictions], as well as all 61 calculated

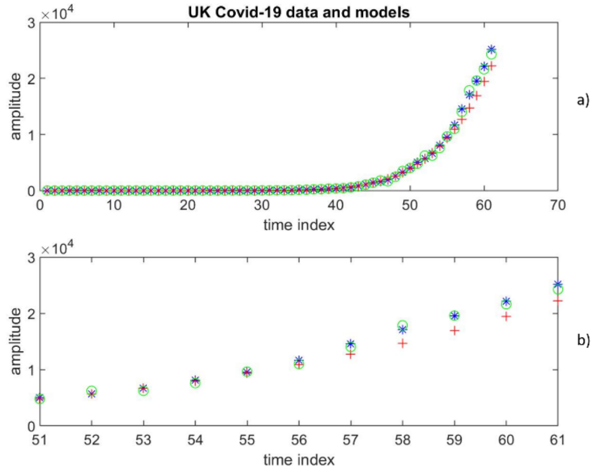

FIGURE 9. Daily confirmed cases of Covid-19 infections in the UK, covering the period from 31 January 2020 to 31 March 2020, Figure 9a) depicts all 61 data values (y) in blue, all 61 calculated values (*yp*) in red according to polynomial representation at $(p + 1) = 6$ [the first 50 values are from the fit and the last 11 values are predictions], as well as all 61 calculated values (*yt*) in green according to autoregressive time-series representation of order 2 [the first 50 values are from the fit and the last 11 values are predictions]. Figure 9b) presents the last 11 data values (y) in blue, the 11 predicted values (*yp*) in red according to polynomial representation at $(p + 1) = 6$, as well as the 11 predicted values (*yt*) in green according to autoregressive time-series representation of order 2.

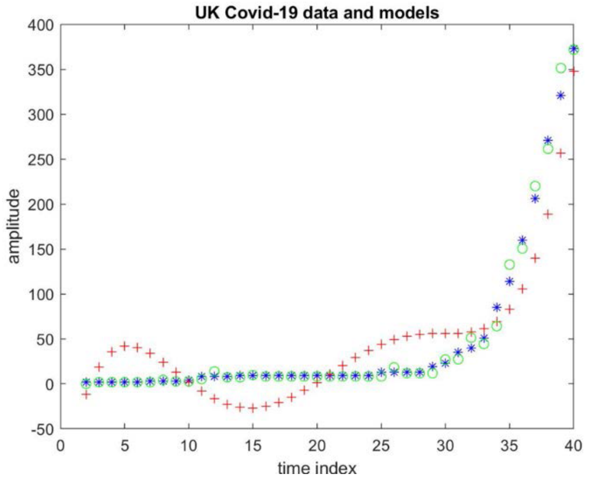

FIGURE 10. Cumulative daily confirmed cases of Covid-19 infections in *the* UK, covering the period from 31 January 2020 to 31 March 2020, Figure 10 plots data values (y) at $n = 2, 3, \ldots, 40$ in blue, the corresponding fitted values (*yp*) in red according to polynomial representation at $(p + 1) = 6$, as well as the corresponding fitted values (*yt*) in green according to autoregressive time-series representation of order 2.

values (yt) in green according to autoregressive time-series representation of order 2 [the first 50 values are from the fit and the last 11 values are predictions]. To get a closer look at the predictions, Figure 9b) depicts the last 11 data values (y) in blue, the 11 predicted values (yp) in red according to polynomial representation at $(p + 1) = 6$, as well as the 11 predicted values (yt) in green according to autoregressive time-series representation of order 2.

To get a better idea of the fit (and not the predictions) Figure 10 plots data values (y) at $n = 2, 3, \ldots, 40$ in blue, the corresponding fitted values (yp) in red according to polynomial representation at $(p + 1) = 6$, as well as the corresponding fitted values (yt) in green according to autoregressive time-series representation of order 2.

It is clear that the polynomial representation picks up the trend of the later data values, but it completely fails for the first half of the data values. On the other hand, this autoregressive time-series of order 2 picks up the trend over the whole range of the data values. The results confirm that the UK Covid-19 data are significantly better described by an AR time-series of order 2 (less RMS error) than a finite degree polynomial of degree 5 (and others).

F. CASE VI

This is another example of using real data. The dataset represents cumulative daily confirmed cases of Covid-19 infections in the US. This dataset is publicly available [39]. On 04 April 2020 there were 25 data (i.e., N = 25) covering the period from 10 March 2020 to 03 April 2020. Thus $y(n)$ for $n = 1 : 1 : 25$ represents the cumulative daily confirmed cases of Covid-19 infections in the US.

Of these 25 data, the first 15 data are used for estimating the free parameters and the last 10 data are used for forecasting. For a polynomial of the degree p, the first 15 data are used to estimate the $(p + 1)$ coefficients of the polynomial using the Moore-Penrose inverse in much the same way as for Case V above. Using these estimated polynomial coefficients, all 25 data are calculated in a similar manner to Case V. The YP is a column vector of size 25×1 and $YP = [yp\,(1)\,yp\,(2)\ldots yp\,(25)]^T$. $[yp\,(1)\,yp\,(2)\ldots yp\,(15)]^T$ came from the regression but $[yp\,(16)\,yp\,(17)\ldots yp\,(25)]^T$ are polynomial predictions for $[y\,(16)\,y\,(17)\ldots y\,(25)]^T$.

Similarly, for a time-series of order q, the first 15 data are used to estimate the q coefficients of the time-series. Each of these data values depends on the coefficients and earlier data values. As all data values are error prone, the Total Least Squares, which takes account of errors in both the dependent and independent variables, is more appropriate than the ordinary Least Squares, which takes account of only errors in dependent variables and not in the independent variables. Using the q estimated coefficients, all 25 data are calculated, i.e., $YT = [yt\,(1)\,yt\,(2)\ldots yt\,(25)]^T$. Of course, out of these 25 values, $[yt\,(1)\,yt\,(2)\ldots yt\,(15)]^T$ came from the regression and $[yt\,(16)\,yt\,(17)\ldots yt\,(25)]^T$ are time-series predictions for $[y\,(16)\,y\,(17)\ldots y\,(25)]^T$.

It is not known a priori whether the data can be represented by a finite degree polynomial or a finite order time-series. The RMS error at $(p + 1) = 4$ is 15272, while the RMS error at $q = 3$ is 6533. Figure 11a) depicts the 25 data values (y) in blue, 25 calculated values (yp) in red according to polynomial representation at $(p + 1) = 4$ [the first 15 values are from the fit and the last 10 values are predictions], as well as 25 calculated values (yt) in green according to autoregressive time-series representation of order 2 [the first 15 values

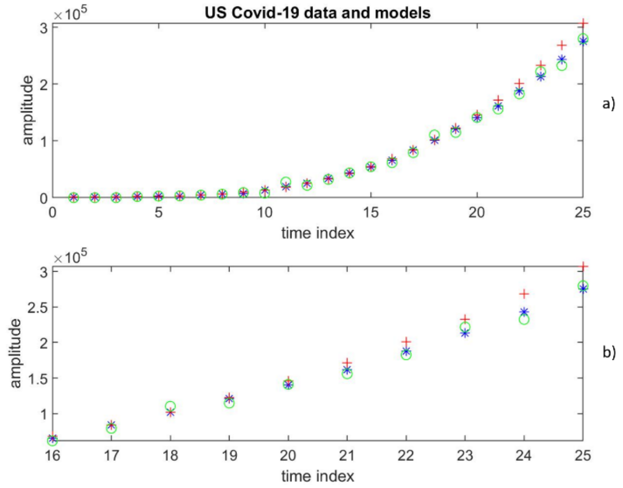

FIGURE 11. Cumulative daily confirmed cases of Covid-19 infections in the US, covering the period from 10 March 2020 to 03 April 2020, Figure 11a) depicts the 25 data values (y) in blue, 25 calculated values (*yp*) in red according to polynomial representation at $(p + 1) = 4$ [the first 15 values are from the fit and the last 10 values are predictions], as well as 25 calculated values (*yt*) in green according to autoregressive time-series representation of order 2 [the first 15 values are from the fit and the last 10 values are predictions]. To get a closer look at the predictions, Figure 11b) shows the last 10 data values (y) in blue, the 10 predicted values (*yp*) in red from polynomial representation at $(p + 1) = 4$, as well as the 10 predicted values (*yt*) in green from autoregressive time-series representation of order 3.

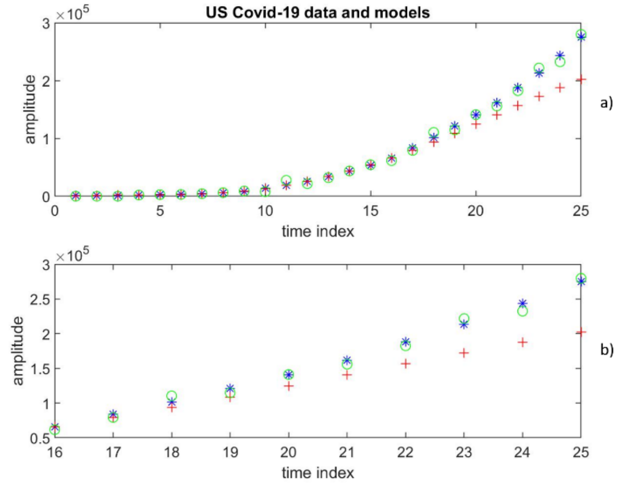

FIGURE 12. Daily confirmed cases of Covid-19 infections in the US, covering the period from 10 March 2020 to 03 April 2020, Figure 12a) and Figure 12b) are for $(p + 1) = 5$; otherwise, they can be similarly described as in Figure 11, except for a different value of $(p + 1)$.

are from the fit and the last 10 values are predictions]. To get a closer look at the predictions, Figure 11b) shows the last 10 data values (y) in blue, the 10 predicted values (yp) in red according to polynomial representation at $(p + 1) = 4$, as well as the 10 predicted values (yt) in green according to autoregressive time-series representation of order 3. RMS errors increase for other choices of $(p + 1)$ values. Clearly, the time-series representation is much more accurate.

Looking for better results with a higher degree of polynomial, the RMS error at $(p + 1) = 5$ is found to be 34692, which is significantly larger than the value from time-series representation at $q = 2$. Figure 12a) and Figure 12b) for $(p + 1) = 5$ can be similarly described as Figure 11 for $(p + 1) = 4$. The results confirm that these US Covid-19 data are significantly better described by an AR time-series of order 3 than a finite degree polynomial of degree 3 (and others).

Table 2 provides a summary of these six cases. Data from a polynomial of finite degree can be represented equally well by a finite degree polynomial as well as a finite order time-series with specific integer coefficients, while data from other sources are represented significantly better by time-series representations. In many cases, finite order time-series can theoretically represent data from infinite order polynomials extremely well. Therefore, whenever the knowledge of the polynomial coefficients is not necessary in an application, one may choose to use time-series representation.

V. TIME-SERIES WITH OTHER COEFFICIENTS

It has been demonstrated in sections II and III that all data from polynomials of finite degree q can be perfectly represented by a time-series of order q, if μ is not zero. The coefficients of such time-series are always integers of a specified form. Below are demonstrated what time-series with other forms of coefficients (either non-integers or integers of different forms) represent.

The equation (2) is called non-homogeneous if μ in equation (2) is not zero [40], [41]. Then equation (2) can be combined with its equivalent form

$$y(n-1) = \sum_{i=1}^{q} a(i)\, y(n-1-i) + \mu \tag{34}$$

to obtain (by replacing μ)

$$y(n) = \sum_{i=1}^{q+1} (a(i) - a(i-1))\, y(n-i) \tag{35}$$

with $a(0) \equiv -1$ and $a(q+1) \equiv 0$. This is a homogeneous equation. The corresponding characteristics polynomial has $(q + 1)$ roots, i.e., $z(1), z(2), \ldots, z(q + 1)$. When these roots are distinct,

$$y(n) = b(1)\, z(1)^n + b(2)\, z(2)^n + \ldots + b(q+1)\, z(q)^n \tag{36}$$

On the other hand, when there are repeated roots, the solution is different. For only two repeated roots, e.g., $z(1) = z(2) = z$,

$$y(n) = b(1)\, z^n + b(2)\, n z^n + b(3)\, z(3)^n + \ldots + b(q+1)\, z(q)^n \tag{37}$$

Each of these two solutions in equations (36) and (37) describes polynomials of infinite degrees. Hence, finite

TABLE 2. Summary of six cases.

Data source	Number of data	Degree of polynomial	Polynomial representation	Time-series representation	Comments
Case I Polynomial	35	3	At $(p+1)=4$, RMS prediction error of 10^{-12}.	At $q=4$, RMS prediction error of 10^{-9}.	Both offer very low prediction errors.
Case II Sine wave	35	∞	At $(p+1)=1$, RMS prediction error of 3.8.	At $q=2$, RMS prediction error of 10^{-15}.	Time-series offers significantly better prediction.
Case III Non-polynomial	35	∞	At $(p+1)=3$, RMS prediction error of 10^{6}.	At $q=4$, RMS prediction error of 10^{-8}.	Time-series offers significantly better prediction.
Case IV Inverse polynomial	35	∞	At $(p+1)=3$, RMS prediction error of 6.7.	At $q=7$, RMS prediction error of 10^{-8}.	Time-series offers significantly better prediction.
Case V UK Covid-19 data	61	Unknown	At $(p+1)=6$, RMS prediction error of 1711.	At $q=2$, RMS prediction error of 539.	Time-series offers much better prediction.
Case VI US Covid-19 data	25	Unknown	At $(p+1)=4$, RMS prediction error of 15272.	At $q=3$, RMS prediction error of 6532.	Time-series offers much better prediction.

order time-series with other forms of coefficients (either non-integers or integers of different forms) represent polynomials of infinite degrees.

A. EXAMPLE I

In case II above, N data were generated from a sine wave

$$y(n) = sin(2\pi n/16), \quad \text{for } n = -17:1:17.$$

Fitting $y(n) = a(1)\,y(n-1) + a(2)\,y(n-2)$ to the first 4 data values, it was found that $a(1) = 1.8478$ and $a(2) = -1.000$. These give rise to the characteristic polynomial of $z^2 - 1.8478z + 1 = 0$. The two roots are given by $z(1) = 0.9239 + 0.d3827j$ and $z(2) = 0.9239 - 0.3827j$. As these two roots are distinct, the solution is given by $y(n) = b(1)\,z(1)^n + b(2)\,z(2)^n$. Since $y(0) = 0$, $b(1) = -b(2)$. Also, since $y(4) = 1$, $b(1) = -0.5j$. So,

$$\begin{aligned} y(n) &= -0.5j\,(0.9239+0.3827j)^n + 0.5j\,(0.9239-0.3827j)^n \\ &= -0.5j\left(\exp\left(\frac{2\pi}{16}\right)\right) + 0.5j\left(\exp\left(-\frac{2\pi}{16}\right)\right)\Big)^n \\ &= sin(2\pi n/16) \end{aligned}$$

This demonstrates how a time-series of order 2 can represent perfectly this sine wave which is a polynomial of infinite degree.

B. EXAMPLE II

Here N data are generated from a non-polynomial

$$y(n) = 1 - 2n - 3n(n-1) + (0.5)^n, \quad \text{for } n = -17:1:17.$$

Fitting $y(n) = a(1)\,y(n-1) + a(2)\,y(n-2) + a(3) * y(n-3) + a(4)\,y(n-4)$ to the first 8 data values, it is found that $a(1) = 3.5, a(2) = -4.5, a(3) = 2.5$, and $a(4) = -0.5$. These give rise to the characteristic polynomial of $z^4 - 3.5z^3 + 4.5z^2 - 2.5z + 0.5 = 0$. The four roots are given by $z(1) = z(2) = z(3) = 1 = z$ and $z(4) = 0.5$. As these are three repeated roots, the solution is given by $y(n) = (b(1) + b(2)\,n + b(3)\,n(n-1))\,(+1)^n + b(4)\,(0.5)^n$. Using the first 8 data values, it can be shown that $b(1) = 1, b(2) = -2, b(3) = -3$, and $b(4) = 1$. Therefore, $y(n) = 1 - 2n - 3n(n-1) + (0.5)^n$.

This is yet another example of how a finite order time-series (in this case of order 4) can represent perfectly this polynomial of infinite degree.

VI. ALL-POLE FILTERS AND POLYNOMIALS

In this section a connection between polynomials and all-pole filters is demonstrated.

A. POLYNOMIALS AND ALL-POLE FILTERS

It is well known that AR time-series models can be realised with all-pole filters. It has already been proven in Section III that all polynomials of finite degree of q can be represented by AR time-series of order q (as in equation (2)). Using z-transform, equation (2) can be written as

$$Y(z) = \mu/(1 - \sum_{i=1}^{q} a(i)\,z^{-i})$$

and it has been proven in Section III that

$$a(i) = (-1)^{i+1}\binom{q}{i}$$

for $i = 1, 2, \ldots, q$. So, the denominator polynomial can now be written as

$$1 - \sum_{i=1}^{q} (-1)^{i+1}\binom{q}{i} z^{-i}$$

$$1 + \sum_{i=1}^{q} (-1)^{i}\binom{q}{i} z^{-i} = \left(1 - z^{-1}\right)^q$$

Therefore, all polynomials of finite degree q map onto $z = 1$ on the z-plane by its q repeated roots.

B. OTHER ROOTS ON THE UNIT CIRCLE

All roots on the unit circle away from $z = -1$ and $z = 1$ are complex. For a real-valued time-series, complex roots come in complex conjugate pairs. Consider just one such pair for illustration, i.e.,

$z(1) = e^{-j\theta}$ and $z(2) = e^{+j\theta}$. Thus, $y(n) = b(1)\,z(1)^n + b(2)\,z(2)^n = b(1)\,e^{-j\theta n} + b(2)\,e^{+j\theta n}$. Since $y(n)$ is real-valued, either $b(1) = b(2) \equiv b$ and $y(n) = 2b\cos(\theta n)$, or $b(1) = -b(2) \equiv -b/j$ and $y(n) = 2b\sin(\theta n)$. Therefore, each pair of complex conjugates roots represent either a cosine or a sine, which can be described by an AR time-series of order 2 instead of a polynomial of infinite degree. The corresponding time-series coefficients are $a(1) = 2\cos(\theta)$ and $a(2) = -1$.

C. OTHER COMPLEX CONJUGATE ROOTS NOT ON THE UNIT CIRCLE

Again, for a real-valued time-series, complex roots come in complex conjugate pairs, consider just one pair for illustration, i.e., $z(1) = \beta e^{-j\theta}$ and $z(2) = \beta e^{+j\theta}$, with $0 < \beta < 1$. In this case, $y(n) = b(1)\,z(1)^n + b(2)\,z(2)^n = b(1)\,\beta^n e^{-j\theta n} + b(2)\,\beta^n e^{+j\theta n}$. Since $y(n)$ is real-valued, either $b(1) = b(2) \equiv b$ and $y(n) = 2b\beta^n\cos(\theta n)$, or $b(1) = b(2) \equiv -b/j$ and $y(n) = 2b\beta^n\sin(\theta n)$. Therefore, each pair of complex conjugates roots represent either a damped cosine or a damped sine, which can be described by an AR time-series of order 2 instead of a polynomial of infinite degree. The corresponding time-series coefficients are $a(1) = 2\beta cos(\theta)$ and $a(2) = -\beta^2$.

D. REAL ROOTS BETWEEN -1 AND +1

Let $-1 < z(1), z(2), z(3) < 1$ be the three distinct roots of the denominator polynomial. Then $y(n) = b(1)\,z(1)^n + b(2)\,z(2)^n + b(3)\,z(3)^n$. This can be described by an AR time-series of order 3 rather than a polynomial of infinite degree.

On the other hand if $-1 < z(1), z(2), z(3) < 1$ be the three repeated roots of the denominator polynomial, i.e., $z(1) = z(2) = z(3) \equiv z$. In that case $y(n) = z^n[b(1) + b(2)\,n + b(3)\,n\,(n-1)]$. This is another example of a finite order AR time-series representing data that requires a polynomial of infinite degree.

The two lessons are:

1) All polynomials of degree q can be represented by an all-pole filter with q repeated roots (or poles) at $z = +1$.
2) Data representable by finite order all-pole filters, whether they are from finite degree or infinite degree polynomials, can be described by a finite order AR time-series.

VII. CONCLUSION

Two of the data modelling techniques are polynomial representation and time-series representation. In this paper, all theoretical studies to explore their connections and differences have been based on uniformly sampled data in the absence of errors. It has been proven that all data from an underlying polynomial model of finite degree q as in equation (21) can be represented perfectly by either a polynomial of degree q or an autoregressive time-series of order q and a constant term. Also, it has been proven that all polynomials of degree q can be described by the same set of time-series coefficients with the only possible difference being in the constant term μ as in equation (2). These time-series coefficients are integers of a specific form. It was also demonstrated that time-series with either non-integer coefficients or integer coefficients of not the specific form represent polynomials of infinite degree. Explorations, in four cases with generated data and in two cases with real data, demonstrated that, while finite degree polynomial and finite order time-series representations are equally good for data following finite degree polynomial forms, finite order autoregressive time-series representations offer significant advantages in modelling data from other sources. All polynomials of degree q can be represented by an all-pole filter with q repeated roots (or poles) at $z = +1$. Theoretically, all data representable by a finite order all-pole filter, whether they come from finite degree or infinite degree polynomials, can be described by a finite order AR time-series. If the values of polynomial coefficients are not necessary in an application, one may choose to use finite order time-series representations as they are more general than finite degree polynomial representations.

ACKNOWLEDGMENT

The author acknowledges Dr C Liu for formatting the manuscript and Dr D A Nandi for supplying the US Covid-19 data.

REFERENCES

[1] A. M. Legendre, *Nouvelles Méthodes Pour la Détermination Des Orbites Des Comètes* (Sur la Méthode Des Moindres Quarrés). Paris, France: Chez Firmin DIDOT, 1805.

[2] C. F. Gauss, *Theoria Combinationis Observationum Erroribus Minimis Obnoxiae*. Gottingen, Germany: Henricus Dieterich, 1823.

[3] J. D. Gergonne, "The application of the method of least squares to the interpolation of sequences," *Historia Math.*, vol. 1, no. 4, pp. 439–447, Nov. 1974, doi: 10.1016/0315-0860(74)90034-2.

[4] S. M. Stigler, "Gergonne's 1815 paper on the design and analysis of polynomial regression experiments," *Historia Math.*, vol. 1, no. 4, pp. 431–439, 1974, doi: 10.1016/0315-0860(74)90033-0.

[5] G. U. Yule, "On the theory of correlation," *J. Roy. Stat. Soc.*, vol. 60, no. 4, pp. 812–854, 1897, doi: 10.2307/2979746.

[6] K. Pearson, G. U. Yule, N. Blanchard, and A. Lee, "The law of ancestral heredity," *Biometrika*, vol. 2, no. 2, pp. 211–236, 1903, doi: 10.2307/2331683.

[7] R. A. Fisher, "The goodness of fit of regression formulae, and the distribution of regression coefficients," *J. Roy. Stat. Soc.*, vol. 85, no. 4, pp. 597–612, 1922, doi: 10.2307/2341124.

[8] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. Cambridge, MA, USA: MIT Press, 1949.

[9] G. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control, Revised Edition*. Oakland, CA, USA: Holden-Day, 1976.

[10] J. Hamilton, *Time Series Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 1994.

[11] N. Gershenfeld, *The Nature of Mathematical Modeling*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[12] W. A. Woodward, H. L. Gray, and A. C. Elliott, *Applied Time Series Analysis*. Boca Raton, FL, USA: CRC Press, 2012.

[13] J. Mandel, *The Statistical Analysis of Experimental Data*. New York, NY, USA: Interscience, 1964.

[14] M. Falk *et al. A First Course on Time Series Analysis: Examples With SAS*. Accessed: Jun. 9, 2020. [Online]. Available: https://www.uni-wuerzburg.de/fileadmin/10040800/user_upload/time_series/the_book/2011-March-01-times.pdf

[15] R. S. Tsay, *Financial Time Series* (Wiley StatsRef: Statistics Reference Online). Hoboken, NJ, USA: Wiley, 2014, pp. 1–23.

[16] C. Goutte, P. Toft, E. Rostrup, F. Å. Nielsen, and L. K. Hansen, "On clustering fMRI time series," *NeuroImage*, vol. 9, no. 3, pp. 298–310, 1999.

[17] F. Mormann, R. G. Andrzejak, C. E. Elger, and K. Lehnertz, "Seizure prediction: The long and winding road," *Brain*, vol. 130, no. 2, pp. 314–333, 2006.

[18] J. M. Craddock, "The analysis of meteorological time series for use in forecasting," *Statistician*, vol. 15, no. 2, p. 167, 1965.

[19] W. Enders, *Applied Econometric Times Series*, 4th ed. Hoboken, NJ, USA: Wiley, 2015.

[20] A. K. Nandi, D. J. Roberts, and A. K. Nandi, "Prediction paradigm involving time series applied to total blood issues data from England," *Transfusion*, vol. 60, no. 3, pp. 535–543, Mar. 2020.

[21] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, no. 4, pp. 561–580, Apr. 1975.

[22] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. Hoboken, NJ, USA: Wiley, 1996.

[23] S. O. Haykin, *Adaptive Filter Theory*, 5th ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2013.

[24] T. S. Rappaport, *Wireless Communications: Principles and Practice*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[25] G. C. Carter, "Coherence and time delay estimation," *Proc. IEEE*, vol. 75, no. 2, pp. 236–255, Feb. 1987.

[26] V. Zarzoso and A. K. Nandi, "Noninvasive fetal electrocardiogram extraction: Blind separation versus adaptive noise cancellation," *IEEE Trans. Biomed. Eng.*, vol. 48, no. 1, pp. 12–18, 2001.

[27] P. Varotsos, N. V. Sarlis, and E. S. Skordas, *Natural Time Analysis: The New View of Time: Precursory Seismic Electric Signals, Earthquakes and Other Complex Time Series*. Secaucus, NJ, USA: Springer, 2011.

[28] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-26, no. 1, pp. 43–49, Feb. 1978.

[29] T. W. Liao, "Clustering of time series data—A survey," *Pattern Recognit.*, vol. 38, no. 11, pp. 1857–1874, Nov. 2005, doi: 10.1016/j.patcog.2005.01.025.

[30] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering—A decade review," *Inf. Syst.*, vol. 53, pp. 16–38, Oct. 2015, doi: 10.1016/j.is.2015.04.007.

[31] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: A survey and empirical demonstration," *Data Mining Knowl. Discovery*, vol. 7, pp. 349–371, Oct. 2003, doi: 10.1023/A:1024988512476.

[32] M. Fahim and A. Sillitti, "Anomaly detection, analysis and prediction techniques in IoT environment: A systematic literature review," *IEEE Access*, vol. 7, pp. 81664–81681, 2019, doi: 10.1109/ACCESS.2019.2921912.

[33] M. Ali, A. Alqahtani, M. W. Jones, and X. Xie, "Clustering and classification for time series data in visual analytics: A survey," *IEEE Access*, vol. 7, pp. 181314–181338, 2019, doi: 10.1109/ACCESS.2019.2958551.

[34] S. BuHamra, N. Smaoui, and M. Gabr, "The Box–Jenkins analysis and neural networks: Prediction and time series modelling," *Appl. Math. Model.*, vol. 27, no. 10, pp. 805–815, Oct. 2003.

[35] X. Wang and C. Wang, "Time series data cleaning: A survey," *IEEE Access*, vol. 8, pp. 1866–1881, 2020, doi: 10.1109/ACCESS.2019.2962152.

[36] L. Zhong, L. Mu, J. Li, J. Wang, Z. Yin, and D. Liu, "Early prediction of the 2019 novel coronavirus outbreak in the mainland China based on simple mathematical model," *IEEE Access*, vol. 8, pp. 51761–51769, 2020, doi: 10.1109/ACCESS.2020.2979599.

[37] I. S. Gradshteyn and I. M. Ryzhik, *Tables of Integrals, Series, and Products*, vol. 10, A. Jeffrey, Ed. New York, NY, USA: Academic, 1980.

[38] *UK Covid-19 Data*. Accessed: Apr. 1, 2020. [Online]. Available: https://www.arcgis.com/home/item.html?id=e5fd11150d274bebaaf8fe2a7a2bda11

[39] *US Covid-19 Data*. Accessed: Apr. 4, 2020. [Online]. Available: https://www.kaggle.com/c/covid19-global-forecasting-week-3/data and https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series

[40] H. Levy and F. Lessman, *Finite Difference Equations*. New York, NY, USA: Dover, 1992.

[41] W. G. Kelly and A. C. Peterson, *Difference Equations: An Introduction With Applications*. Amsterdam, The Netherlands: Elsevier, 2000.

ASOKE K. NANDI (Fellow, IEEE) received the Ph.D. degree in physics from the University of Cambridge (Trinity College), Cambridge, U.K.

He held academic positions at several universities, including the University of Oxford, U.K., Imperial College London, U.K., the University of Strathclyde, U.K., and the University of Liverpool, U.K., as well as the Finland Distinguished Professorship with Jyvaskyla University, Finland. In 2013, he moved to Brunel University London, U.K., to become the Chair and Head of electronics and computer engineering. He is a Distinguished Visiting Professor at Xi'an Jiaotong University, China, and an Adjunct Professor with the University of Calgary, Canada. In 1983, he co-discovered the three fundamental particles known as W^+, W^-, and Z^0 (with the UA1 Team at CERN), providing the evidence for the unification of the electromagnetic and weak forces, for which the Nobel Committee for Physics awarded the prize to his two team leaders for their decisive contributions, in 1984. He has made many fundamental, theoretical, and algorithmic contributions to many aspects of signal processing and machine learning. He has much expertise in Big and Heterogeneous Data. He has authored over 600 technical publications, including 240 journal articles as well as five books, entitled *Automatic Modulation Recognition of Communications Signals* (Springer, 1996), *Blind Estimation Using Higher-Order Statistics* (Springer, 1999), *Automatic Modulation Classification: Principles, Algorithms, and Applications* (Wiley, 2015), *Integrative Cluster Analysis in Bioinformatics* (Wiley, 2015), and *Condition Monitoring With Vibration Signals: Compressive Sampling and Learning Algorithms for Rotating Machines* (Wiley, 2020). The h-index of his publications is 75 (Google Scholar) and his ERDOS number is 2. His current research interests include signal processing and machine learning, with applications to communications, image segmentations, and biomedical data.

Prof. Nandi is a Fellow of the Royal Academy of Engineering, U.K., as well as seven other institutions. He was an IEEE EMBS Distinguished Lecturer from 2018 to 2019. Among the many awards, he received the Mountbatten Premium, the Division Award of the Electronics and Communications Division, Institution of Electrical Engineers, U.K., in 1998, the Water Arbitration Prize of the Institution of Mechanical Engineers, U.K., in 1999, the Glory of Bengal Award for his outstanding achievements in scientific research, in 2010, and the Institute of Electrical and Electronics Engineers (USA) Heinrich Hertz Award, in 2012.